Risk Matrix: An Approach for Identifying, Assessing, and Ranking Program Risks

Paul R. Garvey
Zachary F. Lansdowne

Introduction

Risk Matrix is a structured approach that identifies which risks are most critical to a program and provides a methodology to assess the potential impacts of a risk, or set of risks, across the life of a program. The approach was devised by the acquisition reengineering team at the Air Force Electronic Systems Center (ESC) in 1995. (4) Since January 1996, a number of ESC programs have implemented Risk Matrix.

To facilitate its use, The MITRE Corporation developed a Risk Matrix software application. New analytical features were also added as part of the software development. These include an automated way to cross-check the risk ratings produced by Risk Matrix, as well as an approach for measuring risk mitigation progress. Built in Excel 5.0, the application is cross-platform compatible and can be used on either the Macintosh or PC platforms. This article describes the original Risk Matrix, recently added analytical features, and the software application.

Original Risk Matrix

In Risk Matrix, a risk refers to the possibility that a program's requirement cannot be met by available technology or by suitable engineering procedures or processes. The approach focuses on the requirements-technology pair as the basis for identifying whether a risk exists to the program. A sample Risk Matrix is shown in Table 1. Once a risk (or set of risks) is identified, the subsequent steps in a Risk Matrix are: assess its potential program impacts, hypothesize the probability the risk will occur, rate the risk according to a predetermined scale, and document an action plan to manage/mitigate the risk.

A Risk Matrix is typically completed by a risk management Integrated Product Team (IPT) in a workshop environment. The participants are usually members of the program office and are familiar with the program's technical and programmatic issues, as well as with relevant technologies. They need to work together to identity the program risks and to make the impact and probability assessments. The results are then entered into the Risk Matrix software application, or simply recorded on paper in the appropriate columns. Table 1 illustrates the original Risk Matrix developed in 1995. (4) Each column is defined as follows:

- **Requirements**. List the program's requirements. Typically, these come from two main sources: high-level operational requirements, such as the Operational Requirements Document (ORD), and programmatic requirements, such as those listed in the Program Management Directive (PMD).
- **Technology**. List available technologies that would help meet each requirement. If the technology does not exist or is not mature enough to support the requirement, the probability of a risk occurring becomes higher.

Requirement (Threshold)	Technology	Risk	*I*	P_o %	*R*	Manage/Mitigate
1. VHF Single Channel Communications	ARC-186	• Poor Design	C	0-10	M	• Demonstration as Part of Source Selection
2. Talk SINCGARS	ARC-210 ARC-201 GRC-114	• Algorithm Misunderstood • ICD Problems	C	41-60	H	• Demonstration as Part of Source Selection
3. Talk 100 Miles	ARC-210	• Antenna Performance	S	61-90	M	• Key Parameter of Test Program
4. Go On A-10, F-16, JSTARS and ABCCC	Technology Currently Not Available	• Wrong Power Supply Ratings • Wrong Connectors • Cosite Problems	Mi	0-10	L	• Aircraft Surveys During Ground Team Meeting
5. Control Radio With Control Head	N/A	• Hard to Get Pilot Consensus	Mi	91-100	H	• Control Head Demonstrations Early in Program
6. Joint Program Office	N/A	• Different Users	S	41-60	M	• Information and Decision-Making System
7. Schedule: 24 Months Delivery	N/A	• Integrated Circuit Lead Time	S	11-40	M	• Incentivize On-Time Delivery

Table 1. Sample Risk Matrix Chart

- **Risks**. Identify and describe the risks that might prevent available technology from meeting each requirement.
- **Impact (*I*)**. Assess the impact the risk could have on the program. A default scale is defined in Table 2.
- **Probability of Occurrence (P_o)**. Assess the probability the risk will occur. A default scale is defined in Table 3.
- **Risk Rating (*R*)**. Determine the risk rating (either Low, Medium, or High) by mapping each (I, P_o) pair into the default matrix shown in Table 4.
- **Manage/Mitigate**. The final step is to document the team's strategy to manage/mitigate the risk.

Borda Voting Method

Once a Risk Matrix is populated with a complete set of inputs, questions arise such as: Which risk is most critical? Where should resources be allocated to eliminate the most troublesome areas of the program? Because Table 4 supports only three distinct ratings (High, Medium, or Low), Risk Matrix's original rating method necessarily yields an ordering with many ties. In the case of the sample Risk Matrix chart in Table 1, two risks tie for first place (the High designations), four risks tie for the second place (the Medium designations), and one risk is in third place (the Low designation). In an actual application of Risk Matrix, seven risks tied for first place, thirty-two for second place, and nineteen for third place. With so many ties, it is difficult to isolate the most critical areas of risk from those that are less threatening to the program.

To deal with ties, we incorporated a simple technique from voting theory into the Risk Matrix software application. The technique is known as the Borda method. (2,5,6) When applied to Risk Matrix, the Borda method ranks risks from most to least critical on the basis of multiple evaluation criteria, as described next.

Let N be the total number of risks, which is the same as the number of rows in Risk Matrix. Let the index i denote a particular risk and the index k denote a criterion. The original Risk Matrix

Impact Category	Definition
Critical (C)	An event that, if it occurred, would cause program failure (inability to achieve minimum acceptable requirements).
Serious (S)	An event that, if it occurred, would cause major cost/schedule increases. Secondary requirements may not be achieved.
Moderate (Mo)	An event that, if it occurred, would cause moderate cost/schedule increases, but important requirements would still be met.
Minor (Mi)	An event that, if it occurred, would cause only a small cost/schedule increase. Requirements would still be achieved.
Negligible (N)	An event that, if it occurred, would have no effect on the program.

Table 2. Risk Matrix Impact Assessments (Illustrative Definitions)

Probability Range	Interpretation
0-10%	Very Unlikely to Occur
11-40%	Unlikely to Occur
41-60%	May Occur About Half of the Time
61-90%	Likely to Occur
91-100%	Very Likely to Occur

Table 3. Probability of Occurrence (P_o): Illustrative Interpretations

	Negligible	Minor	Moderate	Serious	Critical
0-10%	Low	Low	Low	Medium	Medium
11-40%	Low	Low	Medium	Medium	High
41-60%	Low	Medium	Medium	Medium	High
61-90%	Medium	Medium	Medium	Medium	High
91-100%	Medium	High	High	High	High

Table 4. Possible Risk Rating Scale (*R*)

has only two criteria: the impact I is denoted by $k = 1$ and the probability assessment P_o is denoted by $k = 2$. If r_{ik} is the rank of risk i under criterion k, the Borda count for risk i is given by

$$b_i = \sum_k (N - r_{ik})$$

Equation 1

The risks are then ordered (ranked) according to these counts. If ties are present in the criteria rankings, the r_{ik} are adjusted by evaluating the rank for a tied alternative as the arithmetic average of the associated rankings. (5,6)

Table 5 is a screen capture that shows how the data in Table 1 appear in the application program. Although Table 5 is very similar to Table 1, a major difference is the new column labeled "Borda Rank." The Borda method is used to aggregate the rankings for I and P_o to obtain an overall ranking for the risks. These results are displayed in the new column. The Borda Rank for a given risk is the number of other risks that are more critical. For example, risk number 2 has a Borda Rank of 0, identifying it as the most critical area of the program. Risk number 7 has a Borda Rank of 5, indicating that there are 5 other risks that are more critical.

The Borda method provides several advantages in this application. First, it generally yields a risk ranking with fewer ties than the risk ratings yielded by Table 4. A tie occurs when two risks have the same rating R (High, Medium, or Low) or the same Borda Rank. For example, Table 1 has two risks with the High rating and four risks with the Medium rating, whereas Table 5 has only two risks with the same Borda Rank. This example shows that the Borda method does not necessarily eliminate all ties.

Second, the Borda method does not require additional subjective assessments beyond the original I and P_o inputs. In contrast, the ratings in Table 4 are based entirely on subjective assessments.

Third, the Borda rank can be used as a cross-check on the Risk Matrix ratings. They are jointly displayed in Table 5 and require the same inputs. The rank orders may differ between the two methods, beyond simply reducing ties. For example, the Borda method gives risk number 3 a higher priority than risk number 5, even though Table 1 suggests that risk number 5 has a higher priority than risk number 3.

Fourth, a sensitivity analysis can be performed on the I and P_o assessments for a given risk. Such an analysis would show what changes are needed to yield a noncritical rank position for a particular risk.

Risk Mitigation Tracking

We also incorporated into the software application an optional method for tracking the progress of risk mitigation actions. The first step is to develop an action plan, composed of a varying number of tasks, to mitigate a given risk. At any point in time, each task in an action plan has a particular status, such as completed or on-track. The second step is to assign one of four colors to represent the status of each task: Blue, Green, Yellow, and Red. The interpretations for these colors are given in Table 6. The third step is to translate each color into the probability that the implementation of the associated task will fail. The default translations are given in Table 6, but they can be changed within the program.

Based upon the color assessment made for each task in an action plan, the fourth step is to evaluate the probability of action plan failure (P_{apf}):

$$P_{apf} = 1 - \prod_j \; [1 - v(y_j)]$$

Equation 2

where y_j is the status color assessed for jth task within the action plan, and $v(y_j)$ is the probability that the implementation of this

Risk No.	Requirement (Threshold)	Technology Available	Risk (to meeting the requirement)	I	Po (%)	Borda Rank	R	Manage/Mitigate
1	VHF Single Channel Comm	ARC-186	Poor design	C	10%	4	M	Demonstration as part of Source Selection
2	Talk SINCGARS	ARC-210, ARC-201, GRC-114	Algorithm misunderstood, ICD problems	C	60%	0	H	Demonstration as part of Source Selection
3	Talk 100 Miles	ARC-210	Antenna performance	S	90%	1	M	Make a key parameter of Test Program
4	Go on A-10, F-16, JSTARS and ABCCC	Technology currently not available	Wrong power supply ratings, wrong connectors, cosite problems	Mi	10%	6	L	Aircraft Surveys during Ground Team Meeting
5	Control Radio with Control Head	N/A	Hard to get pilot consensus	Mi	100%	2	H	Control Head Demonstrations early in
6	Joint Program Office	N/A	Different Users	S	60%	2	M	Information and Decision Making system
7	Schedule: 24 mos. Delivery	N/A	Integrated circuit lead time	S	40%	5	M	Incentivize on-time delivery

Table 5. Risk Matrix Spreadsheet

Color	Interpretation	Default Failure Probability
Blue	The Task Has Been Completed	0.0
Green	The Task Is on Schedule	0.1
Yellow	The Task May Not Be Completed on Schedule	0.5
Red	The Task Is Considered Nonexecutable	1.0

Table 6. Assessment Colors for an Action Plan Task

task will fail. For example, if y_j is yellow, then $v(y_j)$ may be set equal to 0.5. This formula gives the true probability of action plan failure if the tasks are arranged in series and are statistically independent. For a series system, the implementation of the action plan is successful if and only if the implementation of each task within the plan is successful.

It is possible, however, that other circumstances might be present. For example, a set of tasks would form a parallel system when the success of the action plan requires only one of these tasks to be successful. Parallel tasks might be desirable for high-risk exploratory investigations. A given action plan might have a combination of series and parallel tasks. In addition, some tasks might be statistically dependent. Reliability theory has established bounds for these situations. First, if an action plan is coherent (which means that there are no irrelevant tasks), its failure probability cannot exceed the failure probability for all tasks arranged in series. Second, if the tasks are associated (which means they have non-negative covariances), an upper bound on the failure probability of a series system is obtained by treating the tasks as though they were independent. These two results, taken together, imply the above formula provides a rigorous upper bound on the true probability of action plan failure for any set of coherent, associated tasks. (1)

The evaluated probability P_{apf} serves as the measure of risk mitigation progress. The fifth and final step is to rank the risks with the Borda method, but using P_{apf} as one of the criteria instead of the probability of occurrence P_o for each risk having a specified action plan. If an action plan has not been specified for a particular risk, then the program will continue to use P_o as a criterion for that risk.

When applying the foregoing method, the user is responsible for only the first and second steps. After the tasks have been defined and status colors have been assessed, the program automatically carries out the remaining steps.

This tracking method provides several advantages. First, it enables the data and assessments collected for Risk Matrix to be used throughout the risk management process. According to the Defense Systems Management College (DSMC), the risk management process has four basic stages: risk planning, risk assessment, risk analysis, and risk handling. (3) The original Risk Matrix supports the first three stages; the software implementation, with the optional tracking method, supports the fourth stage (measuring risk handling progress).

Second, the risks whose action plans need the closest attention are identified with the Borda method. These critical risks are the ones whose P_{apf} and impact assessment (I) are both relatively high.

Third, if the status colors are assessed periodically (perhaps monthly) for all action plan tasks, both the Borda rank and P_{apf} for each risk can be plotted over time. These high-level graphical displays show the changes in the mitigation status of each risk during the risk-handling stage of the process.

Conclusion

Risk Matrix is a simple, easy to use, structured process that:

- Identifies which risks are most critical to the program, and therefore, most in need of resources.
- Facilitates discussions about requirements, technologies, and risks.
- Allows industry to be involved in the risk assessment and mitigation process early.
- Is a direct way of assessing and managing risk across the life of a program.
- Creates a historical record of program risk and mitigation approaches for deriving lessons learned.
- Is flexible and can be adapted to any project.

The Risk Matrix software application retains all features and capabilities of the original Risk Matrix, without requiring additional steps or data. Its new analytical features include:

- An Excel 5.0/Visual Basic implementation of Risk Matrix compatible on Macintosh and PC platforms.
- An intuitive graphical interface that displays risks by criticality (as illustrated in Table 5).
- Incorporation of the Borda method, a voting algorithm for ranking most-to-least critical risks on the basis of multiple evaluation criteria.
- A method for assessing and tracking risk mitigation action plan progress.
- A way to evaluate the sensitivity of risk rankings to specific evaluation criteria.
- Automatic sorting and charting capabilities.

Risk Matrix is put to use widely at ESC. The software application has been used by both the Joint Surveillance and Target Attack Radar System (JSTARS) and the National Airspace System (NAS) Upgrade program. In the spirit of T. Gilb, using Risk Matrix is one way to "actively attack risks before they actively attack you." (3)

References

1. Barlow, R. E., and F. Proschan, *Statistical Theory of Reliability and Life Testing: Probability Models*, New York: Holt, Reinhart and Winston, 1975.
2. Borda, J-C, "Mémoire sur les Élections au Scrutin," Histoire de l'Académie Royale des Sciences, Paris, 1781.
3. Defense Systems Management College, Risk Management: Concepts and Guidance, Fort Belvoir, Virginia, 1989.
4. Franklin, C. E., Lt Gen, USAF, Commander Electronic Systems Center (ESC), Memorandum for ESC Program Managers, Subject: Risk Management, Department of the Air Force, Headquarters ESC, Air Force Materiel Command, Hanscom Air Force Base, MA, 2 Jan 96.
5. Lansdowne, Zachary F., "Ordinal Ranking Methods for Multicriterion Decision Making," *Naval Research Logistics*, Vol. 43, 1996, pp. 613-627.

CONTINUED ON THE MIDDLE OF PAGE 31

noncommissioned officers with the major systems, maintenance practice, and procedures associated with the F-117 aircraft. The Engine Run Certification course covers ground operation of the F404 engine. It is designed to certify craftsmen-level maintenance technicians in the ground operations of the F404 engines. The Aircraft Mishap Investigation Course (AMIC) teaches the maintenance officer about what characteristics to look for during aircraft mishaps and the causes leading up to the time of the actual mishap. The Jet Engine Mishap Investigation Course (JEMIC) is designed to teach the maintenance officers the same concepts as that of AMIC, but is geared towards jet engines.

All of these courses, except for the ACC "schoolhouse courses," are maintenance related and are geared for the maintenance officers. They are taught at Holloman except for the ACC, AFIT, AMIC, and JEMIC courses. These programs are attended on a temporary duty (TDY) basis and are reserved ahead of schedule to ensure the availability of slots. The ACC schoolhouse course provides an in-depth look at aircraft maintenance concerns. These areas range from flight line organization and leadership to aircraft generation. Two AFIT courses are part of the program curriculum: WLOG 199, Introduction to Logistics, and WLOG 262, Applied Maintenance Management Concepts. WLOG 199 is an entry-level logistics course. WLOG 262 focuses on maintenance management and decision making. All course requirements are outlined in the student LOTP handbook.

All students are provided a LOTP handbook as an information resource. It contains a variety of direct and supplementary information.

LOTP could easily be implemented in other logistics groups. It is adaptable and the 49th Wing has a library of the material used in the program. Other bases and wings can easily tailor the program to meet their need or particular interest. The LOTP is a proven program that helps produce well-rounded logisticians earlier in their careers regardless of Air Force Specialty Code (AFSC).

Lieutenant Rice is presently the Chief, Maintenance Training Flight, 49th Logistics Support Squadron, Holloman AFB, New Mexico.

RETURN TO TABLE OF CONTENTS

CONTINUED FROM PAGE 21

6. Lansdowne, Zachary F., and Beverly S. Woodward, "Applying the Borda Ranking Method," *Air Force Journal of Logistics*, Vol. 20, No. 2 (Spring 96), pp. 27-29.

Paul Garvey and Zachary Lansdowne are presently at the Economic and Decision Analysis Center, The MITRE Corporation, Bedford, Massachusetts.

The authors wish to recognize the fine contributions of the Risk Matrix application development team at The MITRE Corporation. These persons include John A. Wilson, who led the software development, and project contributors James F. Hill (NAS), Catherine W. Kimball (JSTARS), and William L. Parlee (JSTARS).